ISSN 2710-1673. Штучний інтелект, 2020, № 3

УДК 519.6 DOI: https://doi.org/10.15407/jai2020.03.027

О.В. Чистяков
Інститут кібернетики імені В.М. Глушкова НАН України, Україна
пр. Академіка Глушкова, 40, м. Київ, 03187

ПРО ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ МАТЕМАТИЧНОГО МОДЕЛЮВАННЯ ЗАДАЧІ СТІЙКОСТІ КОНСТРУКЦІЙ

A.V. Chistyakov
V.M. Glushkov Institute of Cybernetics, NAS of Ukraine, Ukraine
40, Akademician Glushkov Ave., Kyiv, 03187

ON IMPROVING THE EFFICIENCY OF MATHEMATICAL MODELING OF THE PROBLEM OF STABILITY OF CONSTRUCTION

Анотація. Розглядається алгоритмічно-програмне забезпечення для математичного моделювання стійкості конструкцій, що зводиться до розв'язування часткової узагальненої проблеми власних значень розріджених матриць з автоматичним розпаралеленням обчислень на сучасних паралельних комп'ютерах з графічними процесорами. Представлено особливості реалізації паралельних алгоритмів для різних структур розріджених матриць. Наведено часи розв'язування задачі стійкості композитних матеріалів із застосуванням тривимірної моделі «волокон кінцевих розмірів» на комп'ютерах різної архітектури. При математичному моделюванні фізико-технічних процесів у багатьох випадках виникає потреба розв'язувати задачі алгебраїчної проблеми власних значень (АПВЗ) з розрідженими матрицями великих обсягів. Зокрема, такі задачі виникають при аналізі міцності конструкцій у цивільному та промисловому будівництві, авіабудуванні, електрозварюванні тощо. Розв'язання цих задач полягає у визначенні власних значень і власних векторів розріджених матриць різної структури. Від ефективності розв'язування цих задач у великій мірі залежить ефективність виконання математичного моделювання задачі в цілому. Безперервне зростання параметрів задач, розрахунок на комп'ютерах більш повних моделей об'єктів і процесів потребують зростання продуктивності комп'ютерів. Вимоги до високопродуктивних обчислень набагато випереджають можливості традиційних паралельних комп'ютерів, навіть не зважаючи на багатоядерність процесорів. На сьогоднішній день ця проблема вирішується використанням потужних суперкомп'ютерів гібридної архітектури, наприклад, комп'ютерів з багатоядерними процесорами (CPU) та графічними процесорами (GPU), які поєднують MIMD- і SIMD-архітектури. Але якнайповніше використати потенціал високопродуктивних комп'ютерів можна лише за наявності алгоритмічно-програмного забезпечення, яке враховує як властивості задачі, так і особливості гібридної архітектури. Ускладнення архітектури сучасних високопродуктивних суперкомп'ютерів гібридної архітектури, які активно використовуються для математичного моделювання (збільшення кількості обчислювальних процесорів та ядер, різні види комп'ютерної пам'яті, різні технології програмування тощо) означає істотне ускладнення ефективного використання цих ресурсів при створенні паралельних алгоритмів та програм. Постають проблеми створення алгоритмічно-програмного забезпечення з автоматичним виконанням етапів робіт, які пов'язані з ефективним використанням обчислювальних ресурсів, способами збереження і обробки розріджених матриць, аналізом достовірності комп'ютерних результатів. Це дає можливість значно підвищити ефективність математичного моделювання практичних задач на сучасних високопродуктивних комп'ютерах, а також звільнить користувачів від проблем розпаралелення складних задач. В розробленому алгоритмічно-програмному забезпеченні автоматично реалізуються всі етапи розпаралелення обчислень та обробки розріджених матриць на гібридному комп'ютері. Воно було використано в Інституті механіки імені С.П. Тимошенка НАН України при моделюванні задач міцності композитного матеріалу. Отримано суттєве покращення часових характеристик математичного моделювання. Задачі математичного моделювання властивостей композитних матеріалів відіграють велику роль у проєктуванні процесів деформування та руйнування виробів у різних предметних областях. Розглядається алгоритмічно-програмне забезпечення для математичного моделювання стійкості конструкцій, що зводиться до розв'язування часткової узагальненої проблеми власних значень розріджених матриць різної структури великих порядків з автоматичним розпаралеленням обчислень на сучасних паралельних комп'ютерах з графічними процесорами. Представлено основні методологічні принципи та особливості реалізації паралельних алгоритмів для різних структур розріджених матриць, що забезпечують ефективне виконання багаторівневого паралелізму гібридної системи та зменшують час обміну даних під час обчислювального процесу. Як приклад зазначених підходів, наведено гібридний алгоритм методу ітерації на підпросторі для стрічкових та блочно-діагональних матриць з обрамленням для комп'ютерів гібридної архітектури. Розглянуто особливості декомпозиції даних для матриць профільної структури при реалізації паралельних алгоритмів. Запропонований підхід забезпечує автоматичне визначення необхідної топології гібридного комп'ютера та оптимальну кількість ресурсів для організації ефективного обчислювального процесу. Наведено результати тестування роз-

робленого алгоритмічно-програмного забезпечення для задач з колекції університету Флориди, а також часи розв’язування задачі стійкості композитних матеріалів із застосуванням тривимірної моделі «волокон кінцевих розмірів» на комп’ютерах різної архітектури. Результати свідчать про суттєве покращення часових характеристик розв’язування задач.

Ключові слова: алгебраїчна проблема власних значень; розріджені матриці; гібридні комп’ютери; задачі стійкості конструкцій; композитні матеріали; математичне моделювання.

Abstract. Algorithmic software for mathematical modeling of structural stability is considered, which is reduced to solving a partial generalized eigenvalues problem of sparse matrices, with automatic parallelization of calculations on modern parallel computers with graphics processors. Peculiarities of realization of parallel algorithms for different structures of sparse matrices are presented. The times of solving the problem of stability of composite materials using a three-dimensional model of "finite size fibers" on computers of different architectures are given. In mathematical modeling of physical and technical processes in many cases there is a need to solve problems of algebraic problem of eigenvalues (APVZ) with sparse matrices of large volumes. In particular, such problems arise in the analysis of the strength of structures in civil and industrial construction, aircraft construction, electric welding, etc. The solving to these problems is to determine the eigenvalues and eigenvectors of sparse matrices of different structure. The efficiency of solving these problems largely depends on the effectiveness of mathematical modeling of the problem as a whole. Continuous growth of task parameters, calculation of more complete models of objects and processes on computers require an increase in computer productivity. High-performance computing requirements are far ahead of traditional parallel computing, even with multicore processors. High-performance computing requirements are far ahead of traditional parallel computing, even with multicore processors. Today, this problem is solved by using powerful supercomputers of hybrid architecture, such as computers with multicore processors (CPUs) and graphics processors (GPUs), which combine MIMD and SIMD architectures. But the potential of high-performance computers can be used to the fullest only with algorithmic software that takes into account both the properties of the task and the features of the hybrid architecture. Complicating the architecture of modern high-performance supercomputers of hybrid architecture, which are actively used for mathematical modeling (increasing the number of computer processors and cores, different types of computer memory, different programming technologies, etc.) means a significant complication of efficient use of these resources in creating parallel algorithms and programs. here are problems with the creation of algorithmic software with automatic execution of stages of work, which are associated with the efficient use of computing resources, ways to store and process sparse matrices, analysis of the reliability of computer results. This makes it possible to significantly increase the efficiency of mathematical modeling of practical problems on modern high-performance computers, as well as free users from the problems of parallelization of complex problems. he developed algorithmic software automatically implements all stages of parallel computing and processing of sparse matrices on a hybrid computer. It was used at the Institute of Mechanics named after S.P. Tymoshenko NAS of Ukraine in modeling the strength problems of composite material. A significant improvement in the time characteristics of mathematical modeling was obtained. Problems of mathematical modeling of the properties of composite materials has an important role in designing the processes of deformation and destruction of products in various subject areas. Algorithmic software for mathematical modeling of structural stability is considered, which is reduced to solving a partial generalized problem of eigenvalues of sparse matrices of different structure of large orders, with automatic parallelization of calculations on modern parallel computers with graphics processors. The main methodological principles and features of implementation of parallel algorithms for different structures of sparse matrices are presented, which ensure effective implementation of multilevel parallelism of a hybrid system and reduce data exchange time during the computational process. As an example of these approaches, a hybrid algorithm of the iteration method in subspace for tape and block-diagonal matrices with a frame for computers of hybrid architecture is given. Peculiarities of data decomposition for matrices of profile structure at realization of parallel algorithms are considered. The proposed approach provides automatic determination of the required topology of the hybrid computer and the optimal amount of resources for the organization of an efficient computational process. The results of testing the developed algorithmic software for problems from the collection of the University of Florida, as well as the times of solving the problem of stability of composite materials using a three-dimensional model of "finite size fibers" on computers of different architectures. The results show a significant improvement in the time characteristics of solving problems.

Keywords: algebraic problem of eigenvalues; sparse matrices; hybrid computers; structural stability problems; composite materials; mathematical modeling$ problems of stability of structures.

Вступ

Розробка моделей і методів дослідження властивостей композитних матеріалів відіграють велику роль у проєктуванні процесів деформування та руйнування виробів у різних предметних областях.

У представленій роботі розглядається математичне моделювання задачі визначення докритичного стану та стійкості композитного матеріалу в рамках «волокон кінцевих розмірів» [1, 2].

Складність отримання аналітичних розв’язків таких задач передбачає дослідження тривимірних моделей, використання сучасних чисельних методів та потужних комп’ютерів для їх реалізації.

При цьому найбільші комп'ютерні ресурси витрачаються на розв'язування часткової узагальненої АПВЗ розріджених додатно визначених матриць різної структури, що виникають у результаті дискретизації крайових задач методом скінченних елементів або скінченних різниць.

Обчислені мінімальні власні значення дають можливість визначити величини критичних параметрів стійкості композитного матеріалу при стисненні поверхневим навантаженням.

З метою підвищення ефективності математичного моделювання стійкості композитних конструкцій створено алгоритмічно-програмне забезпечення для дослідження та розв'язування часткової узагальненої АПВЗ розріджених матриць різної структури на основі методу ітерацій на підпросторі для використання на високопродуктивних багатоядерних комп'ютерах MIMD-архітектури з графічними процесорами. Теоретичні аспекти методу ітерацій на підпросторі викладено в роботі [3], а реалізація на MIMD-комп'ютерах представлена в роботі [4].

При створенні паралельних алгоритмів для комп'ютерів зі складною гібридною архітектурою – гібридних алгоритмів – необхідно передбачити, що алгоритм може бути розпаралелений на обчислювальні процеси, які будуть виконуватися на процесорних ядрах CPU (по одному процесу на ядро), кожен з яких має можливість виконувати багатопоточні однотипні обчислення великих обсягів на графічному прискорювачі (GPU). Під процесом розуміють програму, що виконується на ядрі CPU, використовує для своєї роботи частину локальної оперативної пам'яті й здійснює низку операцій прийому/передачі даних для організації інформаційної взаємодії між іншими процесами та процесорами GPU.

Таким чином, розв'язування задач на гібридному комп'ютері передбачає два рівня паралелізму: паралелізм між підзадачами (закінченими частинами алгоритму, які можуть виконуватися незалежно) та паралелізм операцій всередині підзадачі.

При створенні ефективних алгоритмів розв'язування часткової узагальненої АПВЗ розріджених матриць різної структури великих обсягів на таких комп'ютерах необхідно прагнути зменшити кількість пересилок інформації між CPU та GPU, застосовуючи асинхронне виконання великих обчислень, динамічний паралелізм тощо.

Основними методологічними принципами створення даного алгоритмічно-програмного забезпечення є автоматичне виконання таких етапів робіт: попереднє визначення окремих підзадач для їх ефективної реалізації на CPU та GPU; ідентифікація структури вихідної розрідженої матриці та упорядкування дорегулярної структури; розпаралелення обчислень на паралельній архітектурі комп'ютера – побудова ефективної топології з обчислювальних пристроїв гібридного комп'ютера (вузлів, процесорів і ядер CPU, процесорів GPU) та ефективний розподіл даних задачі між цими пристроями; забезпечення синхронізації обчислень і обмінів даними; аналіз достовірності отриманих комп'ютерних результатів.

Ефективна топологія з оптимальної кількості процесорів визначається на основі теоретичних досліджень розроблених алгоритмів та практичних експериментів розв'язування великої кількості тестових задач з матрицями різної структури та обсягів [5, 6].

Ідентифікація структури розріджених матриць здійснюється при використанні штучних нейронних мереж [7].

Постановка задачі розв'язування АПВЗ розріджених додатно визначених матриць

Знайти r мінімальних власних значень і відповідних їм власних векторів задачі

$$Ax = \lambda Bx, \qquad (1)$$

де A, B – симетричні додатно визначені розріджені матриці порядка n.

Для розв'язування цієї задачі застосуємо метод ітерацій на підпросторі. Цей метод є узагальненням методу зворотних ітерацій і полягає в побудові для задачі (1) послідовності підпросторів E_t ($t = 1, 2, ...$), яка збігається до підпростору E_∞, що містить шукані власні вектори [3]. На t-ій ітерації

обчислюється ортогональний базис підпростору E_t і, якщо досягнута необхідна точність наближеного розв'язку, визначаються шукані власні пари.

Таким чином, ітераційний процес ($t = 1, 2, ...$) розв'язування задачі (1) реалізується за такою схемою:

- знаходження розв'язку системи лінійних алгебраїчних рівнянь (СЛАР)
  - $AX_t = Y_{t-1}$; (2)
- обчислення прямокутної матриці
  - $W_t = BX_t$ (3)
- обчислення проєкцій матриць A та B на підпросторі E_t
  - $A_t = X_t^T Y_{t-1} \equiv X_t^T AX_t$; $B_t = X_t^T W_t \equiv X_t^T BX_t$ (4)
- розв'язування повної проблеми власних значень для проєкцій
  - $A_t Z_t = B_t Z_t \Lambda_t$; (5)
- обчислення наступного наближення

$$Y_t = W_t Z_t. \quad (6)$$

Якщо після t ітерацій виконуються умови закінчення ітераційного процесу, наприклад, $\left|\frac{\lambda_i^{(t)} - \lambda_i^{(t-1)}}{\lambda_i^{(t)}}\right| \le \varepsilon$, то проводиться додаткова ітерація із наближеними розв'язками задачі (1), приймаються $\lambda_i^* = \lambda_i^{(t+1)}$, ($i = 1, 2, \ldots, r$) та перші r стовпчиків матриці $X^* = X_{t+1}Z_{t+1}$ (мається на увазі, що власні значення упорядковані за зростанням $0 < \lambda_1 \le \lambda_2 \le \ldots \le \lambda_r \le \ldots$).

Як відомо [3], ітераційний процес збігається лінійно. Причому швидкість збіжності λ_i визначається відношенням λ_q/λ_1, де q − розмір підпростору E_t, що ітерується. У послідовних реалізаціях алгоритму рекомендується вибирати $q = \min(2r, r+8)$.

Оскільки на кожній ітерації виконується розв'язування СЛАР (2) з однією і тією ж матрицею (A), то трикутне розвинення (наприклад, LL^T) цієї матриці виконується один раз до початку ітераційного процесу.

Реалізація гібридного алгоритму методу ітерацій на підпросторі

На основі проведеного аналізу методу ітерацій на підпросторі та враховуючи архітектурні, а також технологічні особливості гібридних комп'ютерів, задачу розв'язування часткової АПВЗ для розріджених симетричних матриць на гібридному комп'ютері розділимо на наступні чотири підзадачі (різної обчислювальної складності), які будуть виконуватися з використанням процесорів CPU та GPU [5].

1. Формування розподіленої між процесами на CPU матриці Y_0 початкових ітерованих векторів, таких, щоб матриця B_1 з (4) була додатно визначеною. Ця операція може ефективно виконуватись на CPU, причому без обмінів даними між процесами, використовуючи, наприклад, запропонований в [4] паралельний алгоритм.
2. Розв'язування СЛАР методом LL^T-розвинення розрідженої симетричної додатно визначеної матриці A (з використанням GPU для багатопоточного виконання матрично-векторних операцій) [5].
3. Ітераційний процес (2) – (6), в якому для кожного $t = 1, 2, \ldots$ обчислення розподілені між процесами на CPU, виконується в такій послідовності:

- для розв'язування СЛАР (2) використовується отримане у попередньому кроці LL^T-розвинення матриці A, тобто розв'язуються дві системи з трикутними матрицями $LV = Y_{t-1}$ та $L^T X_t = V$ (матрично-векторні операції виконуються на GPU);
- обчислення прямокутної матриці $W_t = BX_t$ (3), в залежності від структури матриці B (розріджена або діагональна), розміру ітерованого підпростору, кількості використаних GPU та об'єму їх пам'яті виконуються або тільки на CPU, або з використанням CPU та GPU, або лише на GPU;
- обчислення добутків прямокутних матриць за формулою (4) для формування

проєкцій матриць A та B на підпростір виконуються на GPU, а збір матриць проєкцій (при використанні більше одного GPU) здійснюється кожним процесом на CPU (такий підхід суттєво скорочує об'єм даних, якими обмінюються CPU та GPU; до того ж на GPU обчислення добутків прямокутних матриць можна виконувати асинхронно з іншими обчислювальними операціями або обмінами);

- для розв'язування повної узагальненої АПВЗ (5), враховуючи порівняно невеликий порядок матриць проєкцій, використовується метод Якобі [4] і задача розв'язується кожним MPI-процесом на CPU (у такому випадку відсутня необхідність обмінів даними між CPU та GPU);
- перевірка умов закінчення ітераційного процесу виконується кожним процесом на CPU;
- обчислення (6) нової матриці інтегрованих векторів Y_t (або матриці наближених власних векторів X^*) виконується на GPU у відповідності з розподілом даних (обчислюється підматриця матриці Y_t або X^*), причому немає необхідності в обмінах даними між процесорними пристроями.

4. Дослідження достовірності результатів та обчислення оцінок похибки отриманого розв'язку задачі [4]. Для обчислення оцінок відносної похибки обчислених наближень до власних значень виконуються математичні операції, аналогічні операціям, які виконуються в ітераційному процесі. Для реалізації цих операцій використовуються ті ж гібридні алгоритми, що й в ітераційному процесі.

Як видно з описаного гібридного алгоритму, найбільшу обчислювальну складність має розв'язування системи лінійних алгебраїчних рівнянь (2) з розрідженою додатно визначеною матрицею, яка може бути різної структури. Ефективність реалізації цієї підзадачі на гібридному комп'ютері у великій мірі залежить від способів збереження елементів матриць та їх розподілу між процесорними пристроями.

Тут розглядаються два варіанти гібридного алгоритму методу ітерацій на підпросторі – для стрічкових симетричних матриць [5] та для блочно-діагональних матриць з обрамленням [6].

Особливості реалізації гібридного алгоритму методу ітерацій на підпросторі для стрічкових матриць

Для застосування гібридного алгоритму розв'язування системи лінійних алгебраїчних рівнянь з симетричною додатно визначеною стрічковою матрицею було проведено розподіл матриці A на квадратні блоки A_{ij} порядка s.

Блоком A_{ij} вважається підматриця матриці A, що знаходиться на перетині рядків $(i-1)s+1, is$ та стовпчиків $(j-1)s+1, js$. Для зручності представлення будемо вважати, що порядок матриці n та напівширина стрічки k кратні s. Нехай $q=\dfrac{n}{s}$, $l=\dfrac{k}{s}$. Тоді блочне представ-лення матриці A має вигляд

$$\begin{pmatrix} A_{11} & 0 & 0 & \ldots & \ldots & 0 \\ A_{21} & A_{22} & 0 & \ldots & \ldots & 0 \\ A_{31} & A_{32} & \ddots & & & 0 \\ \vdots & \vdots & & & & \vdots \\ A_{(l+1)1} & A_{(l+1)2} & & & & \\ 0 & A_{(l+2)2} & & & & \\ 0 & 0 & & & & \\ \vdots & \vdots & & \ldots & A_{(q-1)(q-1)} & 0 \\ 0 & 0 & & & A_{q(q-1)} & A_{qq} \end{pmatrix}$$

Елементи головної діагоналі та нижнього або верхнього трикутника (в залежності від використаного алгоритму) ненульових блоків цієї симетричної матриці розподіляються між процесами CPU у відповідності з одномірною блочно-циклічною схемою [4]. За цією схемою блок A_{ij} зберігається в процесі з логічним номером $(I+l)\bmod p$ (результат операції $k \bmod j$ – залишок від ділення k на j, $-1 \le l \le p-2$ – зсув, зазвичай $l=-1$).

У результаті LL^T – розвинення (2), аналогічно будуть розподілені блоки отриманих нижньої трикутної матриці L або верхньої трикутної матриці L^T.

Така ж блочно-циклічна схема розподілу використовується для елементів матриці *B* та отриманих прямокутних матриць інтегрованих векторів X_t, Y_t, W_t за (3) – (6). При цьому достатньо розподіляти та зберігати лише ненульові елементи матриці *B* в такій послідовності: піддіагональні, діагональні та наддіагональні. Це значно спрощує виконання множення такої матриці на прямокутну на GPU, несуттєво збільшуючи загальний об'єм даних. В пам'яті GPU зберігаються копії тих блоків (відповідно до розподілу між CPU) матриць *A*, *B*, X_t, Y_t, W_t, які використовуються для виконання математичних операцій.

Особливості реалізації гібридного алгоритму методу ітерацій на підпросторі для додатно визначених блочно-діагональних матриць з обрамленням

Структуру блочно-діагональної матриці з обрамленням представлено на рис. 1.

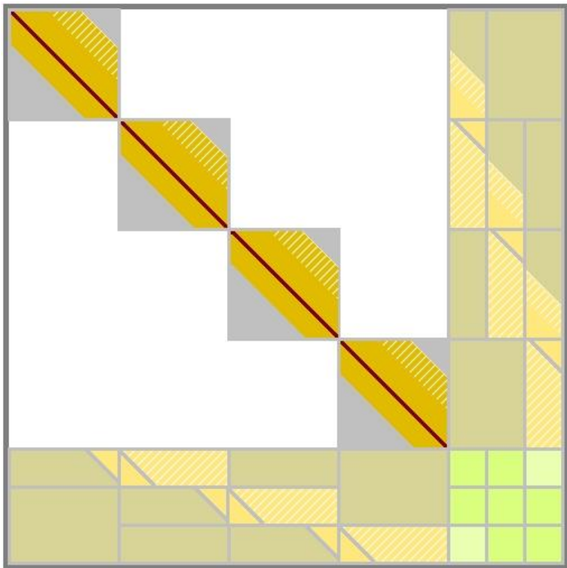


Рис. 1. Коефіцієнт прискорення

У цьому випадку в гібридному алгоритмі для розв'язування АПВЗ-методом ітерацій на підпросторі [6] передбачається, що вихідна матриця може спочатку бути представлена в блочно-діагональному вигляді з обрамленням (у результаті застосування методу скінченних різниць, скінченних елементів або отримана після структурної регуляризації, наприклад, за методом паралельних перерізів, у такому вигляді

$$A = P^T \tilde{A} P = \begin{pmatrix} D_{11} & 0 & 0 & \dots & 0 & C_{1p} \\ 0 & D_{22} & 0 & \dots & 0 & C_{2p} \\ 0 & 0 & D_{33} & & 0 & C_{3p} \\ \vdots & \vdots & & \ddots & & \vdots \\ 0 & 0 & 0 & & D_{p-1p-1} & C_{p-1p} \\ C_{p1} & C_{p2} & C_{p3} & \dots & C_{pp-1} & D_{pp} \end{pmatrix}$$

де *P* – матриця перестановок, а блоки D_{ii}, C_{ip} та C_{pi} зберігають розріджену структуру, *p* – кількість діагональних блоків ($p \geq 3$). Зауважимо, що порядок останнього діагонального блоку може бути набагато меншим, ніж порядки інших діагональних блоків.

Для розв'язування СЛАР за формулою (2) було використано гібридний алгоритм методу LL^T –розвинення блочно-діагональної матриці з обрамленням [6]. Така структура матриці дає можливість блочно-циклічним способом розподілити її між процесами CPU та провести на них паралельну (незалежну) обробку блоків D_i, C_i, $i = \overline{1, p-1}$, задіюючи GPU для матричних операцій. При цьому можна змінювати кількість та порядки блоків у розподілі, що дозволяє гнучко управляти навантаженням на CPU та GPU й забезпечувати хороше балансування між ними.

Крім того, така блочна структура дозволяє працювати з нерозривними масивами даних на GPU, що зменшує кількість індексних операцій і перевірок, які на графічних прискорювачах є досить витратними. Причому кількість блоків слід вибирати з урахуванням кількості процесорів, задіяних у розрахунках.

Експериментальне дослідження гібридних алгоритмів методу ітерацій на підпросторі для розв'язування часткової АПВЗ розріджених матриць

Для дослідження гібридних алгоритмів методу ітерацій на підпросторі використовувалися симетричні розріджені матриці різної структури та різних порядків, у тому числі було використано матриці з колекції університету Флориди [8].

Програмна реалізація гібридних алгоритмів виконана на алгоритмічній мові C++ з використанням системи MPI [9] для розпаралелення на багатоядерних CPU-процесах та технології CUDA [10] для реалізації обчислень на GPU. Виконання операцій матрично-векторної обробки на CPU здійснювалось при використанні відповідних функцій з бібліотеки програм Intel MKL [11], а на GPU – функцій бібліотек програм cuBLAS [12] та cuSparse [13].

Комп'ютерні експерименти проводились на багатоядерному комп'ютері з графічними процесорами з такими технічними характеристиками: CPU серії Intel(R) Xeon(R) CPU E5606, тактова частота 2.13 GHz, швидкість 4,8 GT/s, кеш-пам'ять 8 MB, у вузлі: 2 CPU по 4 ядра, Max Memory Size 288 GB.

На графіках (рис. 2) демонструються прискорення гібридного алгоритму, які отримано при розв'язуванні АПВЗ для різних матриць: Dubcova3 – порядок матриці 146 689, кількість ненульових елементів 3 636 643; bmwcra_1 – порядок матриці 148 770, кількість ненульових елементів 10 641 602; Bone010 – порядок матриці 986 703, кількість ненульових елементів 47 851 783.

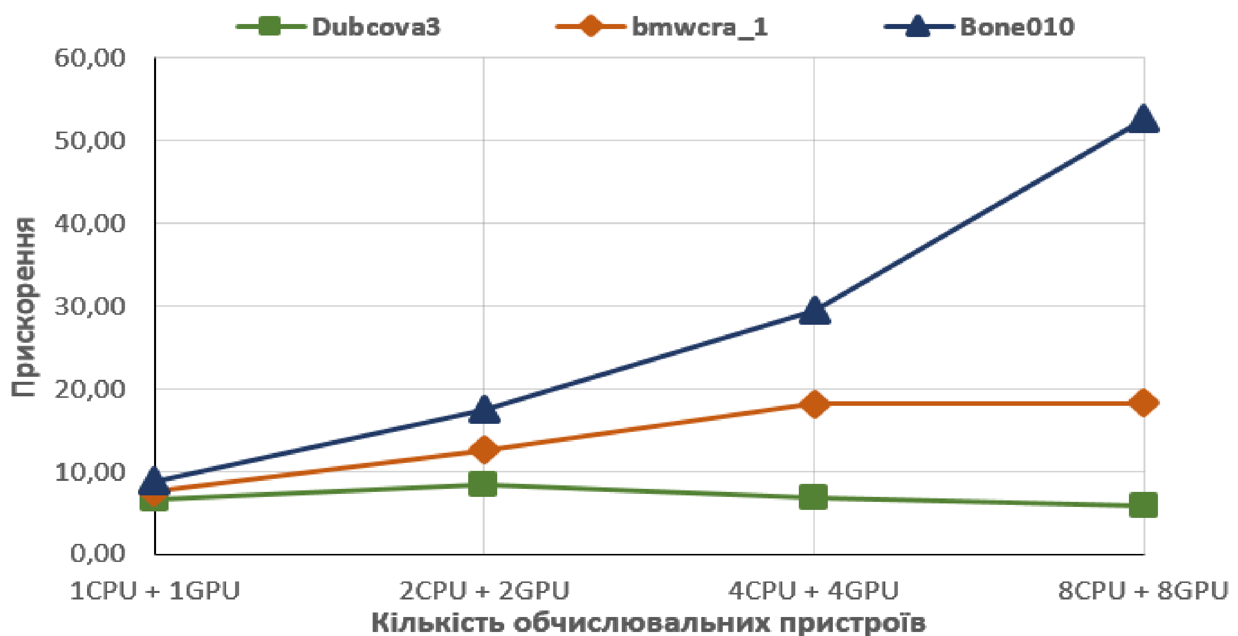


Рис. 2. Прискорення алгоритму для стрічкових симетричних матриць

З рисунку ми бачимо, що гібридний алгоритм методу ітерацій на підпросторі для симетричних стрічкових матриць добре масштабований, прискорення при збільшенні кількості процесів CPU та відповідних процесорів GPU значно зростає для всіх задач.

Крім того, можна відзначити, що для матриць невеликого розміру (Dubcova3) при виході за межі обчислювального вузла (коли кількість використаних CPU та

GPU більша двох) наступає насичення процесу обчислювальними ресурсами. Водночас, для матриць великого порядку bmwcra_1 та Bone010 зі збільшенням кількості CPU та GPU прискорення зростає.

Результати досліджень гібридного алгоритму методу ітерацій на підпросторі для блочно-діагональних матриць з обрамленням наведено на рис. 3. Використано розріджені матриці з флоридської колекції [8].

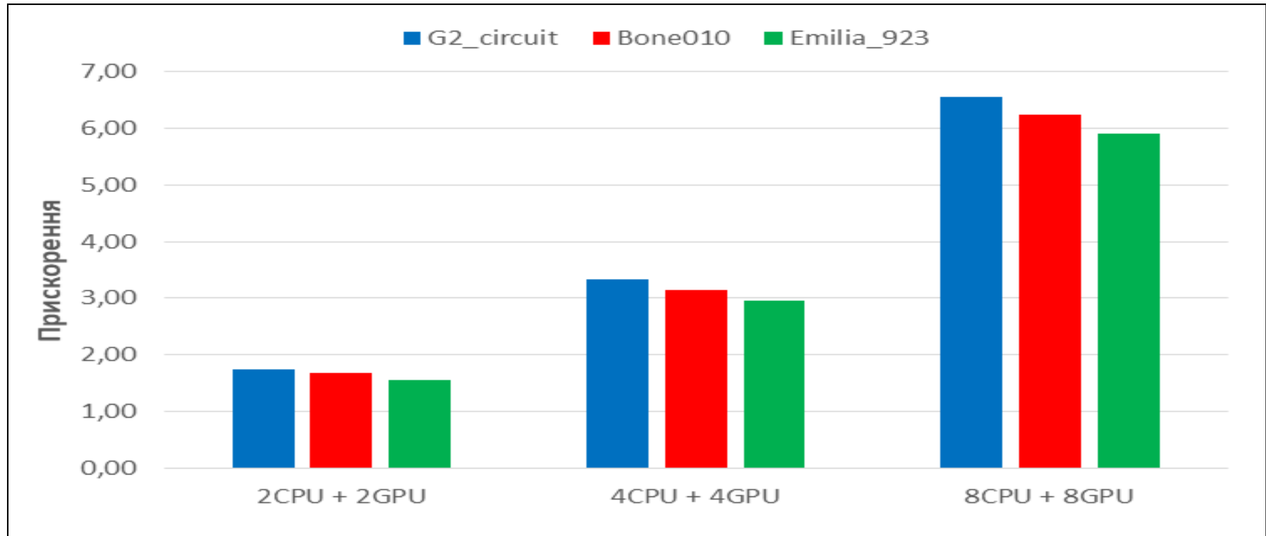


Рис. 3. Прискорення алгоритму для блочно-діагональних матриць з обрамленням

Результати математичного моделювання задачі стійкості композитних матеріалів на гібридному комп'ютері

Задача була розв'язана при використанні нових гібридних алгоритмів розв'язування АПВЗ для розріджених матриць з різними вихідними даними [14].

Нижче наведено результати з такими вихідними даними: порядок матриць – 12282; напівширина стрічки матриці *A* – 6212 та матриці *B* – 71; обсяг пам'яті – 2 Gb.

Для порівняльної характеристики створених гібридних алгоритмів та програм розв'язування проводились на комп'ютерах різної архітектури:

- комп'ютер гібридної архітектури СКІТ-4 з CPU – серії Intel(R) Xeon(R) E5-2600, у вузлі – 2 CPU по 8 ядер, графічні процесори – Nvidia Tesla M2075;
- гібридний комп'ютер СКІТ-4.5 AI з процесорами Intel(R) Xeon(R) E5-2695 v4, у вузлі – 2 CPU по 18 ядер, графічні процесори – Nvidia Tesla K20.

Результати розв'язування задачі при використанні різних версій алгоритму методу ітерацій на підпросторі для додатно визначених стрічкових матриць:

- послідовним алгоритмом час розв'язування – 22 хв. 38 сек;
- паралельним алгоритмом на СКІТ-4, використовуючи 2 вузли (4 процесори по 8 ядер), час розв'язування – 2 хв. 13 сек;
- гібридним алгоритмом на СКІТ-4: при використанні одного CPU та GPU – 1 хв. 30 сек, при використанні двох CPU та GPU – 0 хв. 47 сек;
- гібридним алгоритмом на СКІТ-4.5 AI: при використанні одного CPU та GPU – 0 хв. 35 сек.

Отже, за розробленим гібридним алгоритмом методу ітерацій на підпросторі розв'язування АПВЗ для додатно визначених стрічкових матриць отримано такі прискорення, у порівнянні з послідовною версією алгоритму: на MIMD-комп'ютері – в 9 раз; на гібридному комп'ютері – в 15 і 28 разів, використовуючи один і два GPU відповідно; на гібридному комп'ютері СКІТ-4.5 AI – в 38 раз.

Час розв'язування цієї задачі на СКІТ-4.5 AI за версією гібридного алгорит-

му методу ітерацій на підпросторі для додатно визначених блочно-діагональних матриць з обрамленням становить 7,84 сек.

Розв'язок цієї задачі за допомогою пакета MATLAB [15] було отримано за 340 сек. Отже, за гібридним алгоритмом методу ітерацій на підпросторі для блочно-діагональних матриць з обрамленням задача була розв'язана приблизно в 43 рази швидше, у порівнянні з часом розв'язування за MATLAB та приблизно в 5 раз швидше у порівнянні з часом розв'язування за гібридним алгоритмом призначеного для додатно визначених стрічкових матриць.

Висновки

Запропоновано алгоритмічно-програмне забезпечення на основі методу ітерацій на підпросторі для розв'язування часткової узагальненої проблеми власних значень розріджених матриць довільної структури на багатоядерному комп'ютері з графічними процесорами. Використання нових гібридних алгоритмів та програм у математичному моделюванні задачі міцності композитного матеріалу забезпечує значне прискорення обчислень при ефективному використанні комп'ютерних обчислювальних засобів паралельного комп'ютера зі складною архітектурою.

Література

1. Guz A.N. (1999). *Fundamentals of the Three-Dimentional Theory of Stability of Deformable Bodies.* Berlin–Heidelberg–New York: Springer. DOI: 10.1007/978-3-540-69633-9.
2. Guz A.N, Dekret V.A., Kokhanenko Yu V. (2000). *Solution of plane problems of the three-dimension problems stability of a ribbon-reinforced composite.* Int. Appl. Mech., 36(10). 1317–1328. DOI: 10.1023/A:1009434116426.
3. Парлетт Б. (1983). *Симметричная проблема собственных значений.* М.: Мир, 384.
4. Химич А.Н., Молчанов И.Н., Попов А.В., Чистякова Т.В., Яковлев М.Ф. (2008). *Параллельные алгоритмы решения задач вычислительной математики.* Киев: Наукова думка.
5. A.N. Khimich, A.V. Popov, O.V. Chistyakov. (2017). *Hybrid Algorithms for Solving the Algebraic Eigenvalue Problem with Sparse Matrices.* Cybernetics and Systems Analysis. 53(6). 937–949. DOI: 10.1007/s10559-017-9996-5.
6. Химич А.Н., Попов А.В., Сидорук В.А., Чистяков А.В. (2020). *Параллельный алгоритм решения частичной проблемы собственных значений для блочно-диагональных матриц с окаймлением.* Кибернетика і системний аналіз. 6, 61–74.
7. Сидорук В.А., Єршов П.С., Богурський Д.О., Марочканич О.Р. (2019). *Інтелектуалізація обчислень для задач математичного моделювання складних процесів і об'єктів.* Комп'ютерна математика. 1, 143–150.
8. The SuiteSparse Matrix Collection. Отримано з https://cise.ufl.edu/research/sparse/matrices/
9. Немнюгин С.А., Стесик О.Л. (2002). *Параллельное программирование для многопроцессорных вычислительных систем.* СПб.: БХВ-Петербург.
10. Боресков А.В., Харламов А.А. (2010). *Основы работы с технологией CUDA.* М.: Пресс.
11. Math Kernel Library. Отримано з https://software.intel.com./en-us/mkl/.
12. cuBLAS. Отримано з: https://developer.nvidia.com/cublas/
13. cuSparse Library. Отримано з http://docs.nvidia.com/cuda/cuSPARSE/
14. Khimich A.N., Dekret V.A., Popov A.V., Chistyakov O.V. (2018). *Numerical Study of the Stability of Composite Materials on Computers of Hybrid Architecture.* Journal of Automation and Information Sciences 50 (7). Begell House Inc., 7–24. DOI: 10.1007/s10559-020-00311-z.

References

1. Guz A.N. (1999). *Fundamentals of the Three-Dimentional Theory of Stability of Deformable Bodies.* Berlin–Heidelberg–New York: Springer. DOI: 10.1007/978-3-540-69633-9.
2. Guz A.N, Dekret V.A., Kokhanenko Yu V. (2000). *Solution of plane problems of the three-dimension problems stability of a ribbon-reinforced composite.* Int. Appl. Mech., 36(10). 1317–1328. DOI: 10.1023/A:1009434116426.
3. Parlett B. (1983). *Simmetrichnaya problema sobstvennyih znacheniy*. M.: Mir, 384.
4. Himich A.N., Molchanov I.N., Popov A.V., Chistyakova T.V., Yakovlev M.F. (2008). *Parallelnyie algoritmyi resheniya zadach vyichislitelnoy matematiki*. Kiev: Naukova dumka.
5. A.N. Khimich, A.V. Popov, O.V. Chistyakov. (2017). *Hybrid Algorithms for Solving the Algebraic Eigenvalue Problem with Sparse Matrices.* Cybernetics and Systems Analysis. 53(6). 937–949. DOI: 10.1007/s10559-017-9996-5.
6. Himich A.N., Popov A.V., Sidoruk V.A., Chistyakov A.V. (2020). *Parallelnyiy algoritm resheniya chastichnoy problemyi sobstvennyih znacheniy dlya blochno-diagonalnyih matrits s*

okaymleniem. Kibernetika I sistemniy analIz. 6, 61–74.
7. Sydoruk V.A., Yershov P.S., Bohurskyi D.O., Marochkanych O.R. (2019). *Intelektualizatsiia obchyslen dlia zadach matematychnoho modeliuvannia skladnykh protsesiv i obiektiv.* Kompiuterna matematyka. 1, 143–150.
8. The SuiteSparse Matrix Collection. Retrieved from https://cise.ufl.edu/research/sparse/matrices/
9. Nemnyugin S.A., Stesik O.L. (2002). *Parallelnoe programmirovanie dlya mnogoprotsessornyih vyichislitelnyih sistem.* SPb.: BHV-Peterburg.
10. Boreskov A.V., Harlamov A.A. (2010). *Osnovyi rabotyi s tehnologiey CUDA*. M.: Press.
11. Math Kernel Library. Retrieved from https://software.intel.com./en-us/mkl/.
12. cuBLAS. Retrieved from https://developer.nvidia.com/cublas/
13. cuSparse Library. Retrieved from http://docs.nvidia.com/cuda/cuSPARSE/
14. Khimich A.N., Dekret V.A., Popov A.V., Chistyakov O.V. (2018). *Numerical Study of the Stability of Composite Materials on Computers of Hybrid Architecture.* Journal of Automation and Information Sciences 50 (7). Begell House Inc., 7–24. DOI: 10.1007/s10559-020-00311-z.

Стаття надійшла до редакції 22.07.2020
Після доробки 31.08.2020